Superpolynomial Lower Bounds Against Low-Depth Algebraic Circuits

Nutan Limaye
IIT Bombay $\longrightarrow$ ITU Copenhagen
Joint work with

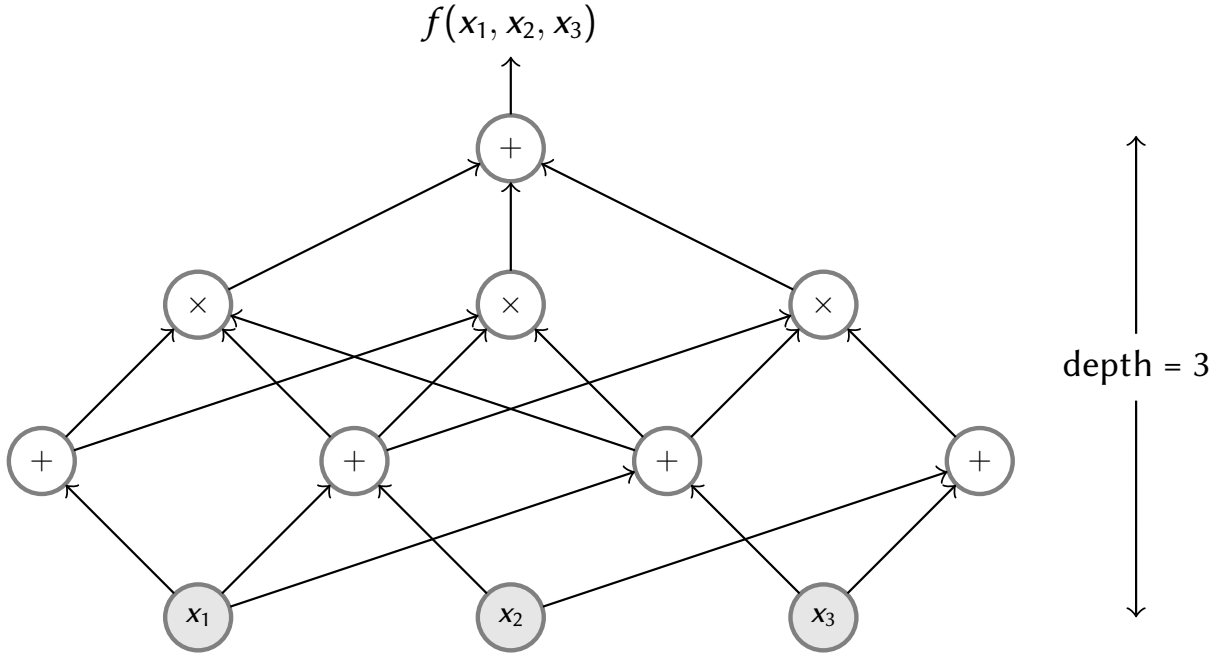
Srikanth Srinivasan (Aarhus University)
Sébastien Tavenas (Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA)

July 9, 2021
Oxford-Warwick Complexity Meetings

Introduction

Let $P(x_1, \ldots, x_N) \in \mathbb{F}[x_1, \ldots, x_N]$ be a polynomial.

An algebraic circuit is a model of computation which computes polynomials.

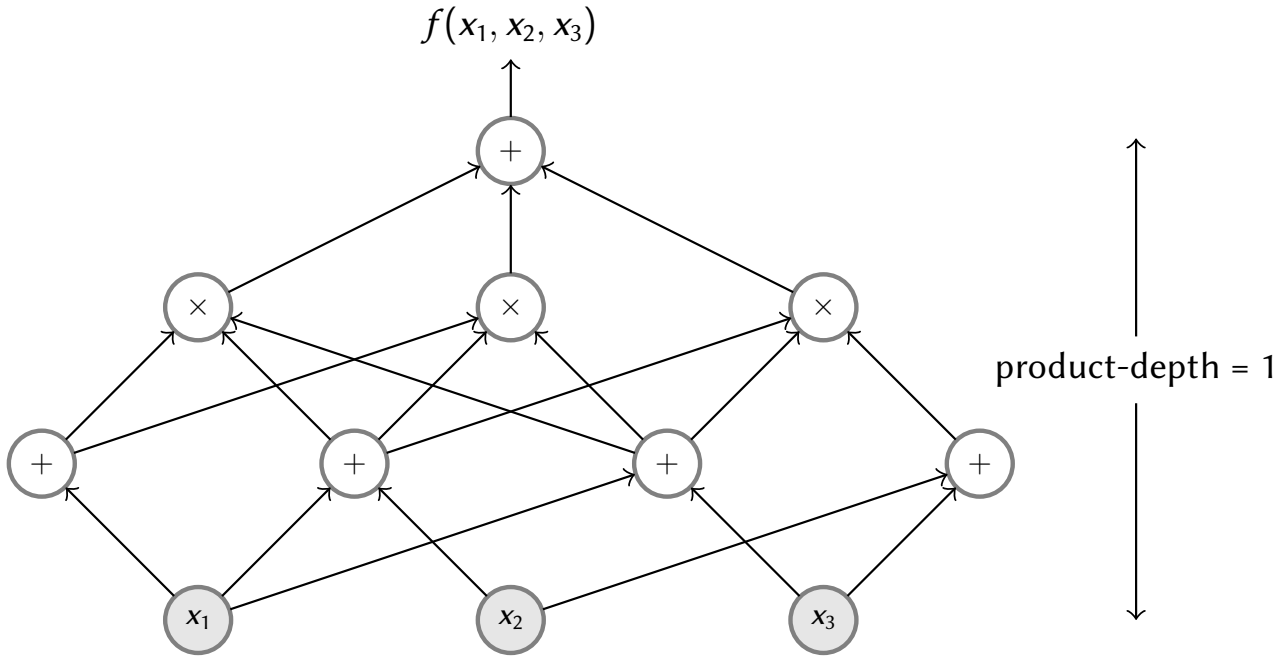


Called $\sum \prod \sum$ circuits.

Introduction

Let $P(x_1, \ldots, x_N) \in \mathbb{F}[x_1, \ldots, x_N]$ be a polynomial.

An algebraic circuit is a model of computation which computes polynomials.



Size = Number of operations. In this case 8.

A formula is a circuit with tree as the underlying DAG.

Algebraic Complexity Theory

Study of polynomials.

- $P(x_1, \ldots, x_N) = \sum_{S \subseteq [N]} \prod_{i \in S} x_i$. Uses $O(2^N)$ operations.
- $P(x_1, \ldots, x_N) = \prod_{i \in [N]} (1 + x_i)$. Uses $O(N)$ operations.

How many operations are needed for computing a polynomial?

Algebraic Complexity Theory

Lower bounds for algebraic circuits

- Algorithm design and lower bounds are duals of each other.
- A step towards P vs. NP
  - Algebraic circuits syntactic objects computing formal polynomials.
  - Algebraic circuit lower bounds formally easier than Boolean circuit lower bounds.

Algebraic circuit lower bounds

Boolean circuit lower bounds.

Strong lower bounds for constant-depth Boolean circuits known since the 80s.

[Ajtai 83, FSS 84, Håstad 86, Razborov 86, Smolensky 87].

Algebraic circuit lower bounds.

The best known lower bound for $\sum\prod\sum$ circuits is $\Omega(N^3/\log^2 N)$ [KST2016].

The best known lower bound for $\sum\prod\sum\prod$circuits is $\Omega(N^{2.5})$ [GST2020].

No superpolynomial lower bounds even for product-depth 1 circuits.

Our Main Result

The first superpolynomial lower bound for constant-depth algebraic circuits.

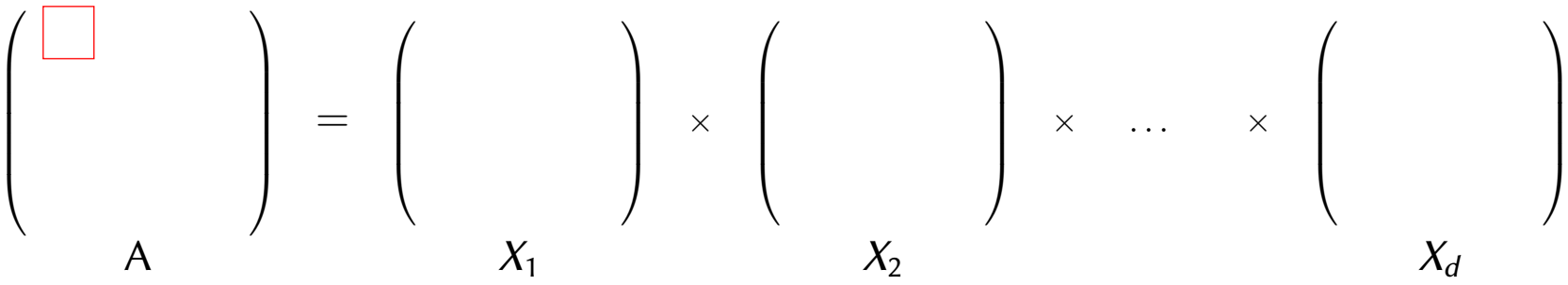
Main Theorem

Let N, d, Δ be growing parameters with $d = o(\log N)$. Assume $\mathbb{F}$ is characteristic 0.

There is an explicit polynomial $P_{N,d}(x_1, \ldots, x_N)$ that any algebraic circuits of product-depth Δ computing it must have size $N^{d^{\exp(-O(\Delta))}}$.

Explicit Polynomial $P_{N,d}$

The hard polynomial is $\mathrm{IMM}_{n,d}$.



$\mathrm{IMM}_{n,d}$ is defined over variable sets $X_1, \ldots, X_d$, each of size n^2.

Each X_i thought of as an $n \times n$ matrix.

$\mathrm{IMM}_{n,d}$ is the $(1,1)$th entry of product $X_1 \cdot X_2 \cdot \ldots \cdot X_d$.

Other Results

Polynomial Identity Testing

Subexponential time PIT

Given black-box access to a constant-depth poly(N)-size circuit C computing a polynomial $P(x_1, \ldots, x_N)$ over characteristic 0, there is a deterministic algorithm for checking whether $P \equiv 0$ that runs in subexponential time.

Other Results

Polynomial Identity Testing

Subexponential time PIT

Given black-box access to a constant-depth poly(N)-size circuit C computing a polynomial $P(x_1, \ldots, x_N)$ over characteristic 0, there is a deterministic algorithm for checking whether $P \equiv 0$ that runs in subexponential time.

Prior to this deterministic $n^{O(k)}$ time algorithm known for $\sum^{[k]} \prod \sum$ circuits. [SS 2012]

Algebraic hardness vs. randomness (by [CKS 2018]) + our lower bound.

[CKS 2018] builds on [KI 2004], [DSY 2009].

Other Results

Polynomial Identity Testing

Subexponential time PIT

Given black-box access to a constant-depth poly(N)-size circuit C computing a polynomial $P(x_1, \ldots, x_N)$ over characteristic 0, there is a deterministic algorithm for checking whether $P \equiv 0$ that runs in subexponential time.

Depth Hierarchy Theorem

Depth Hierarchy

Assume fields of characteristic 0. For every constant $\Gamma \geq 2$ and growing parameter s , there is a polynomial Q_Γ of depth Γ and size s such that any depth $(\Gamma - 1)$ circuit computing it must have size $s^{\omega(1)}$.

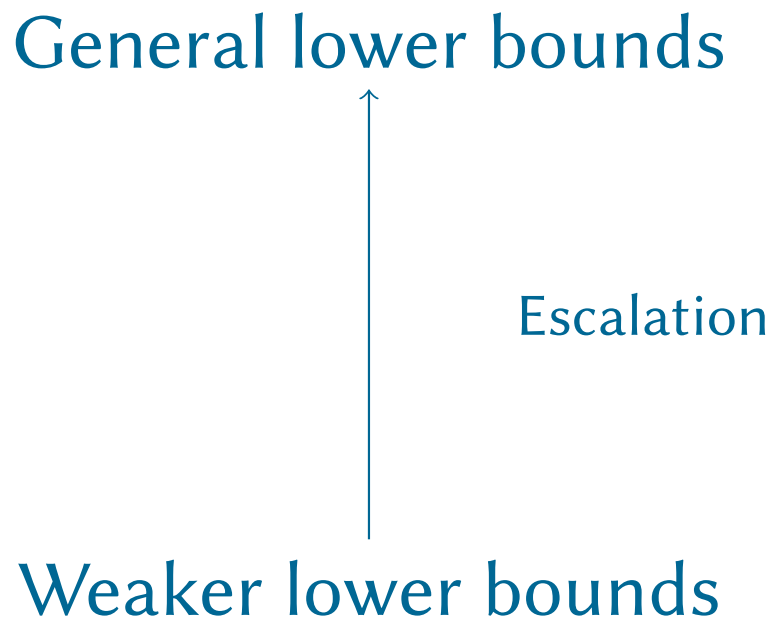
Our Main Result

The first superpolynomial lower bound for constant-depth algebraic circuits.

Main Theorem

Let N, d, Δ be growing parameters with $d = o(\log N)$. Assume $\mathbb{F}$ is characteristic 0.

There is an explicit polynomial $P_{N,d}(x_1, \ldots, x_N)$ that any algebraic circuits of product-depth Δ computing it must have size $N^{d^{\exp(-O(\Delta))}}$.


General lower bounds
Escalation
Weaker lower bounds

Restricted Classes of Polynomials

Homogeneous, multilinear and set-multilinear polynomials

- A polynomial $P(x_1, \ldots, x_N)$ is homogeneous if every monomial in it has the same degree.

- A polynomial is called multilinear if every monomial has at most one occurrence of any variable.

- Let the variable set be partitioned into sets $(X_1, X_2, \ldots, X_d)$.

  A polynomial is called set-multilinear with respect to the partition $(X_1, \ldots, X_d)$ if every monomial has exactly one variable from each set.

  $\mathrm{IMM}_{n,d}$ is a set-multilinear polynomial with variable partition $(X_1, \ldots, X_d)$.

Restricted Models of Computation

Restricted classes of *formulas*.

- A formula is called homogeneous if every gate in the circuit computes a homogeneous polynomial.
- A multiliear formula is defined in the same way.
- A formula is set-multilinear if every gate in the formula computes a set-multilinear polynomial in a subset of $X_1, \ldots, X_d$.

Hardness Escalation Results

Let $P(x_1, \ldots, x_N)$ be a set-multilinear polynomial of degree d.

[Raz 2009]

Formula of size s computing P

Efficient conversion

Set-multilinear formula computing P
of size $\mathrm{poly}(s) \cdot (\log s)^{O(d)}$

Hardness Escalation Results

Let $P(x_1, \ldots, x_N)$ be a set-multilinear polynomial of degree $d = O(\log N / \log \log N)$.

[Raz 2009]

Formula of size $\mathrm{poly}(N)$ computing P

Efficient conversion

Set-multilinear formula computing P of size $\mathrm{poly}(N)$

Hardness Escalation Results

Let $P(x_1, \ldots, x_N)$ be a set-multilinear polynomial of degree $d = O(\log N / \log \log N)$.

[Raz 2009]

Any formula computing P needs superpolynomial size

Escalation

Set-multilinear formula computing P needs size $N^{\omega_d(1)}$

Non-FPT Lower Bounds

Known lower bounds

Known set-multiliear formula lower bounds for constant product-depth. [NW 95, Raz 2009, RY 2009]

$$\Omega(f(d) \cdot \mathrm{poly}(N))$$

For escalation to work, we need.

$$N^{\Omega(f(d))}$$

We call these non-FPT bounds.

Our Non-FPT Lower Bound

Our non-FPT lower bound for set-multilinear formulas.

Set-multilinear formula lower bound

Let $d \leq O(\log n)$. For any $\Delta \geq 1$ any set-multilinear formula C computing $\mathrm{IMM}_{n,d}$ of product-depth Δ must have size $n^{d^{\exp(-O(\Delta))}}$.

Efficient Conversion

Let $P(x_1, \ldots, x_N)$ be a set-multilinear polynomial of degree d.

General Formula
size s depth Δ

Char 0

Homogeneous formula
size $\text{poly}(s) \cdot 2^{O(d)}$ depth 2Δ

Any Char

Set-multilinear formula
size $\text{poly}(s) \cdot d^{O(d)}$ depth 2Δ

Escalation

Let $P(x_1, \ldots, x_N)$ be a set-multilinear polynomial of degree d.

General formula
needs superpolynomial size Δ depth

↑ Char 0

Homogeneous formula
needs size $N^{\omega_d(1)}$ depth 2Δ

↑ Any Char

Set-multilinear formula
needs size $N^{\omega_d(1)}$ depth 2Δ

Our Non-FPT Lower Bounds + Escalations

Our non-FPT lower bound for set-multilinear formulas.

Set-multilinear formula lower bound

Let $d \leq O(\log n)$. For any $\Delta \geq 1$ any set-multilinear formula C computing $\mathrm{IMM}_{n,d}$ of product-depth Δ must have size $n^{d^{\exp(-O(\Delta))}}$.

Let $P(x_1, \ldots, x_N)$ be a set-multilinear polynomial of degree d.

Efficient conversions

General Formula
size s depth Δ

↓ Char 0

Homogeneous formula
size $\mathrm{poly}(s) \cdot 2^{O(d)}$ depth 2Δ

↓ Any Char

Set-multilinear formula
size $\mathrm{poly}(s) \cdot d^{O(d)}$ depth 2Δ

Escalation

General formula
needs superpolynomial size Δ depth

↑ Char 0

Homogeneous formula
needs size $N^{\omega_d(1)}$ depth 2Δ

↑ Any Char

Set-multilinear formula
needs size $N^{\omega_d(1)}$ depth 2Δ

Non-FPT Lower Bounds

Our non-FPT lower bound for set-multilinear formulas.

Set-multilinear formula lower bound

Let $d \leq O(\log n)$. For any $\Delta \geq 1$ any set-multilinear formula C computing $\mathrm{IMM}_{n,d}$ of product-depth Δ must have size $n^{d^{\exp(-O(\Delta))}}$.

- At $\Delta = 2$, that is for $\sum\prod\sum\prod\sum$ set-multilinear formulas, we get a the first tight $n^{\Omega(\sqrt{d})}$ lower bound for $\mathrm{IMM}_{n,d}$.

- For $\Delta = 1$ we get a lower bound of $n^{\Omega(\sqrt{d})}$ for general formulas. Tighness of [GKKS 2014] depth reduction.

Techniques

A typical lower bound proof

The lower bound proof outline.

- Come up with a measure $\mu : \mathbb{F}[x_1, \ldots, x_N] \to \mathbb{R}$.
- Show that $\mu(\mathrm{IMM}_{n,d})$ is large.
- Show that $\mu(\text{s. m. } \sum\prod\sum\prod\sum)$ is small.

We will prove that $\mu(\text{s. m. } \sum\prod\sum\prod\sum)$ is small.

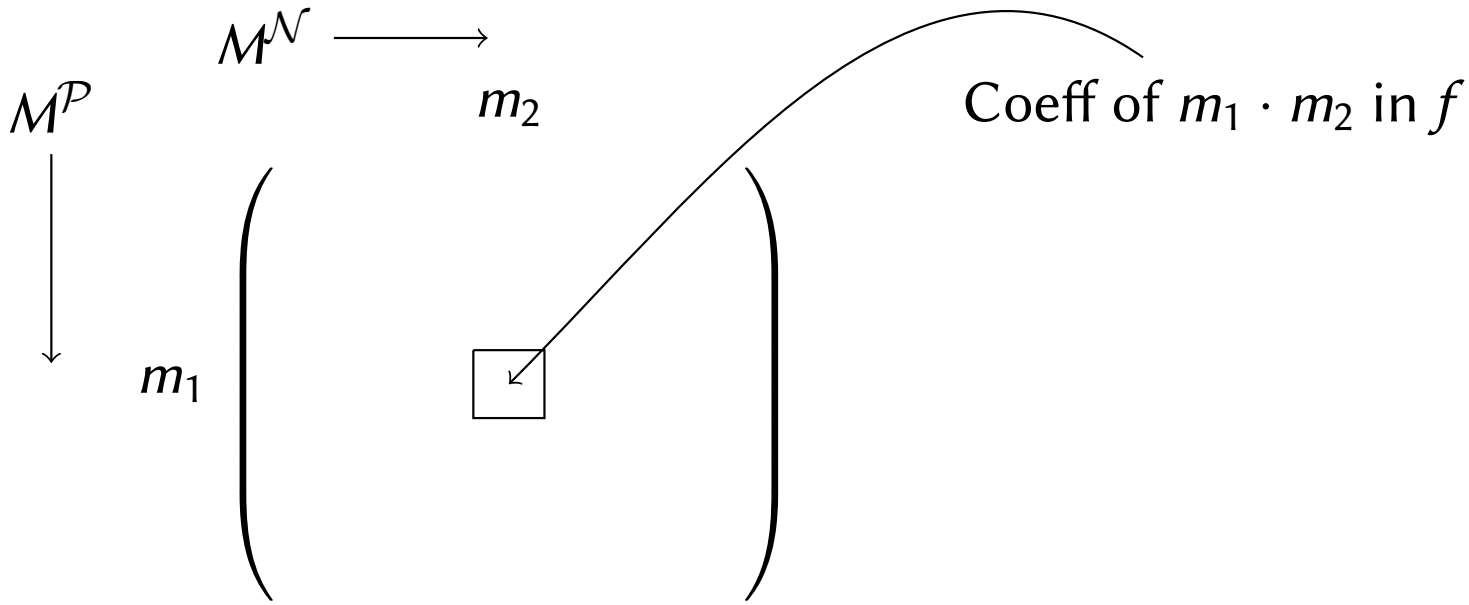
Partial Derivative Measure

Nisan and Wigderson [NW 95]

Partition $[d]$ into $\mathcal{P}$ and $\mathcal{N}$.

$M^{\mathcal{P}}$ multilinear monomials over $(X_i : i \in \mathcal{P})$.
$M^{\mathcal{N}}$ multilinear monomials over $(X_i : i \in \mathcal{N})$.

For a polynomial f, define matrix M_f as follows.



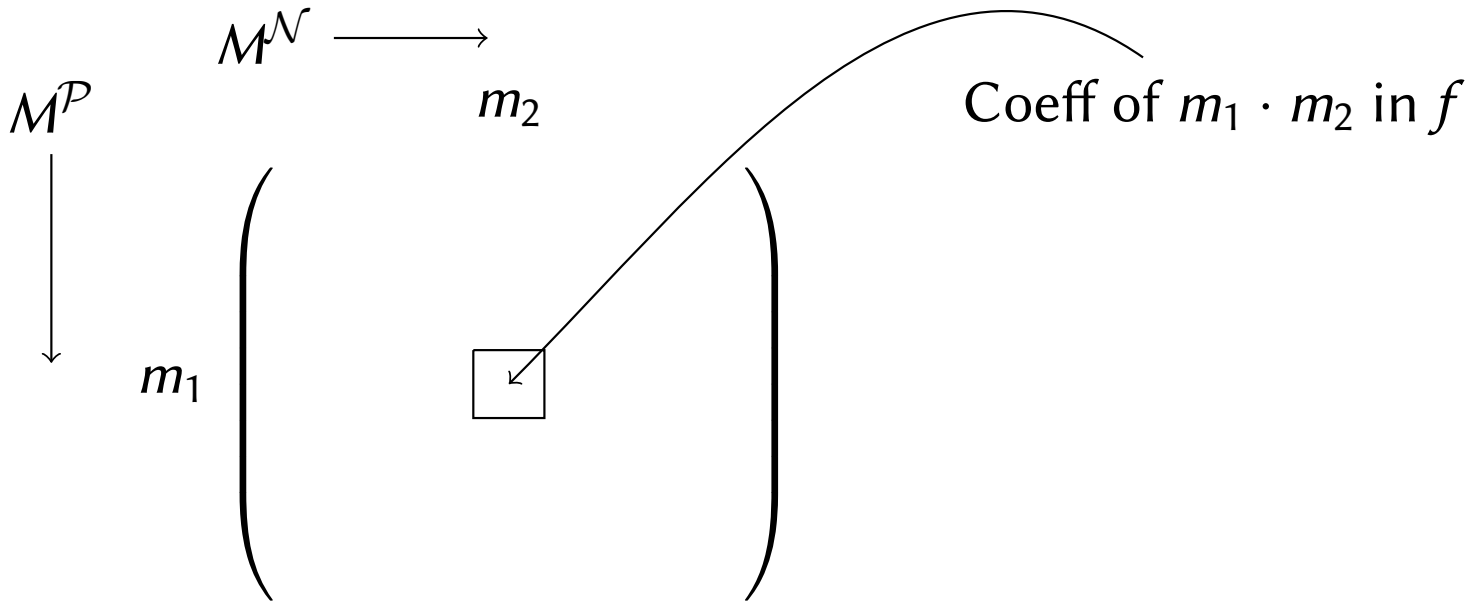
Partial Derivative Measure

Nisan and Wigderson [NW 95]

Partition $[d]$ into $\mathcal{P}$ and $\mathcal{N}$.

$M^{\mathcal{P}}$ multilinear monomials over $(X_i : i \in \mathcal{P})$.
$M^{\mathcal{N}}$ multilinear monomials over $(X_i : i \in \mathcal{N})$.

For a polynomial f, define matrix M_f as follows.



- The Partial Derivative Measure is the $\mathrm{rank}(M_f)$. Denoted $\mathrm{rk}(f)$.

$\sum\prod\sum$ set-multilinear formulas

Let $(X_1, \ldots, X_d)$ be a partition of variables.

$$F(X) = \sum_{i=1}^{s} \prod_{j=1}^{d} \ell_{i,j}(X_j)$$

each $\ell_{i,j}$ homogeneous linear polynomial over X_j.

For each $i \in [s], j \in [d]$, $\mathrm{rk}(\ell_{i,j}(X_j))$ at most 1.

For each $i \in [s]$, $\mathrm{rk}\left(\prod_{j=1}^{d} \ell_{i,j}(X_j)\right)$ at most 1.

By subadditivity of rank, $\mathrm{rk}(F(X))$ at most s.

For $\mathcal{P} = \{i \mid i \text{ odd}\}$ and $\mathcal{N} = \{i \mid i \text{ even}\}$,
$\mathrm{rk}(\mathrm{IMM}_{n,d}) = n^{\Omega(d)}$.

$$s \geq n^{\Omega(d)}$$

$\Sigma\Pi\Sigma\Pi$ set-multilinear formulas

Product of Inner Products Polynomial.

Let $X_j = \{x_{j,1}, \ldots, x_{j,m}\}$ for $j \in [d]$.

$$\mathrm{PIP}(X_1, \ldots, X_d) = \prod_{j=1}^{d/2} \left(\sum_{k=1}^{m} x_{2j-1,k} \cdot x_{2j,k} \right)$$

PIP has product-depth 2 set-multilinear formula of size $O(md)$.

For $\mathcal{P} = \{i \mid i \text{ odd}\}$ and $\mathcal{N} = \{i \mid i \text{ even}\}$,
M_{PIP} is a permutation matrix.

$\mathrm{rk}(\mathrm{PIP}) = m^{\Omega(d)}$.

$\sum\prod\sum\prod$ set-multilinear formulas

Shifted Partial Derivative Measure [Kayal 2012]

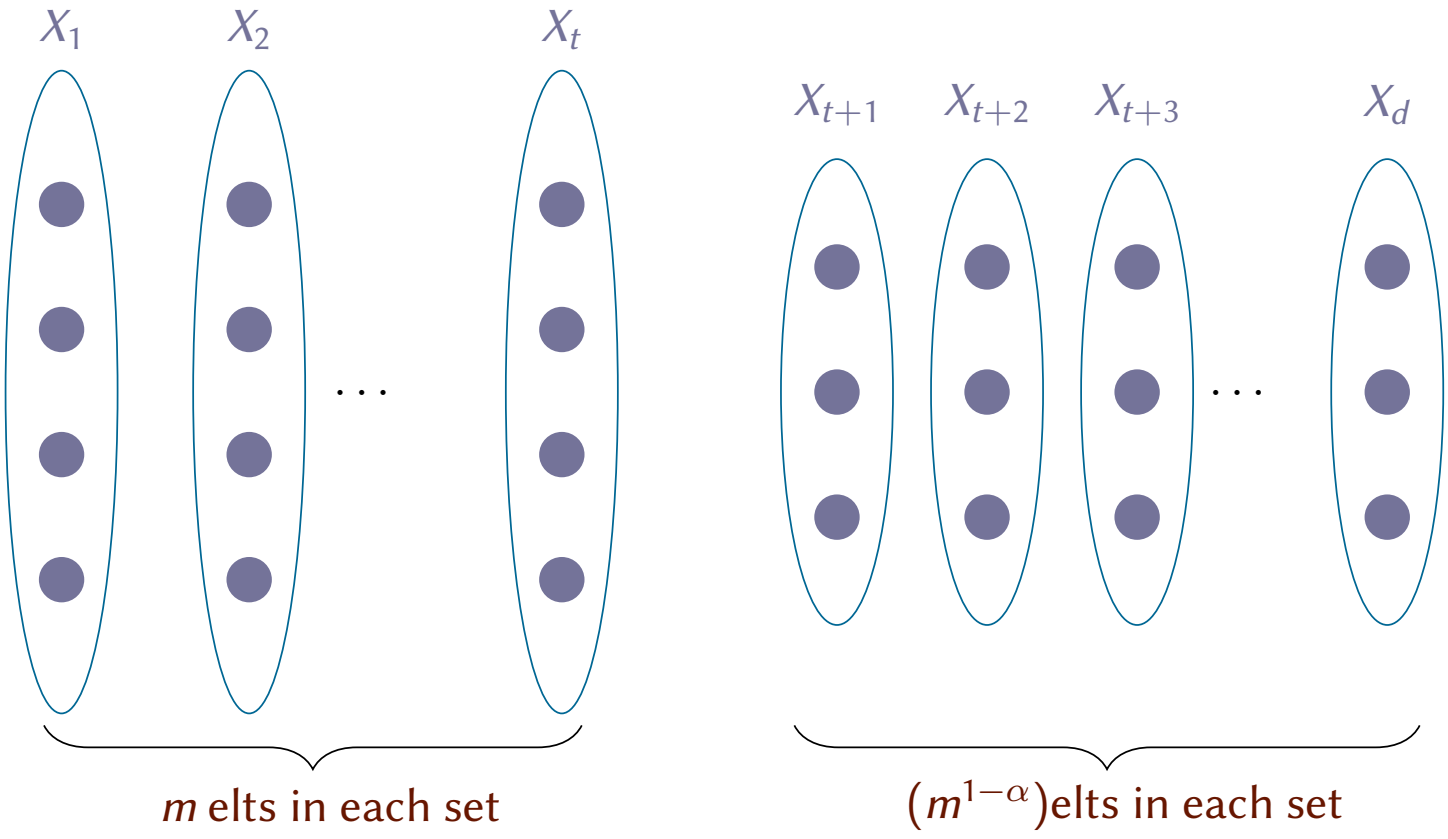
Any set-multilinear $\sum\prod\sum\prod$ formula computing $\mathrm{IMM}_{n,d}$ must have size $n^{\Omega(\sqrt{d})}$.

[GKKS 2014, FLMS 2014, KS 2014, KLSS 2014].

Not clear how to use it to prove lower bounds for $\sum\prod\sum\prod\sum$ set-multilinear formulas.


PARTIAL DERIVATIVES
SHIFTED PARTIALS
PARTIAL DERIVATIVES + DIFFERENT SET SIZES
imgflip.com

Different set sizes



$\mathcal{P} = \{1, 2, \ldots, t\}$ and $\mathcal{N} = \{t+1, t+2, \ldots, d\}$.

For $\alpha = 1/\sqrt{d}$ and t chosen to ensure $|M^{\mathcal{P}}| = |M^{\mathcal{N}}|$

$\Sigma\Pi\Sigma\Pi\Sigma$ set-multilinear formulas

$$\sum \boxed{\prod_j \underbrace{(\Sigma\Pi\Sigma)}_{F_j}}^{F}$$

Focus on one term F, which is $F_1 \times F_2 \times \ldots \times F_k$.

Each F_j is a $(\Sigma\Pi\Sigma)$ set-multilinear formula.

We show that for any term F

- Case 1 Either one of F_js has sufficiently low rank.
- Case 2 All F_js have moderately low ranks.

Then use sub-additivity of ranks.

$\Sigma\Pi\Sigma\Pi\Sigma$ set-multilinear formulas

$$F$$

$$\boxed{\Pi_j \underbrace{(\Sigma\Pi\Sigma)}_{F_j}}$$

Case 1 There is an F_j with degree $\geq \sqrt{d}/2$.

Use the low-rankness of $(\Sigma\Pi\Sigma)$.

Case 2 All F_js have degree $< \sqrt{d}/2$.

We get an imbalance between # rows and # columns in M_{F_j}!
(Thanks to choice of α.)

Choice of α

Simple intuition.

Suppose $A = \{-1, -1, \ldots, -1\}$ and
$B = \{(1 - \frac{1}{p}), (1 - \frac{1}{p}), \ldots, (1 - \frac{1}{p})\}$.
Let $S \subset A \cup B$ such that $|S| < p/2$.

For any such S, the summation of its elements never be 0.

As the fractional part will not cancel for small $|S|$.

This results in an imbalance, which helps in rank upper bounds.

Choice of α for $\Delta = 2$

Choice of α for $\Delta = 2$.

Crucial when in Case 2.

$F = F_1 \times F_2 \times \ldots \times F_k$

Degree of each F_j is at most $\sqrt{d}/2$.

Say M_{F_j} has m^r rows and $m^{(1-\alpha)\cdot c}$ columns.

Here, r and c are integers and $r + c < \sqrt{d}/2$.

$\therefore |r - ((1-\alpha)\cdot c)| = \Omega(\alpha \cdot c)$ as long as $\alpha = 1/\sqrt{d}$.

That is, M_{F_j} is not a square matrix.

Choice of α for higher depths

Choice of α for higher Δ.

We choose $\alpha = 1/\sqrt{2}$ in this case.

This is motivated by Diophantine Approximations.

A similar case analysis as in the $\Delta = 2$ works out.

$d \longrightarrow \sqrt{d} \longrightarrow d^{1/4} \ldots$.

Can handle depths up to $o(\log \log d)$.

Open Questions

Can the lower bound be improved? What about $n^{\Omega(d^{1/\Delta})}$?

Can we improve the escalation? For example, can we remove the Char 0 restriction?

The lower bound neatly gave an efficient PIT. Can we get anything for reconstruction?

Can we get algebraic proof system lower bounds?

Can combining known measures give better lower bounds?

Thank You!